


```

LL          IIIII
LL          IIIII
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LLLLLLLLLL IIIII
LLLLLLLLLL IIIII
SSSSSSSS
SSSSSSSS
SS
SS
SS
SS
SS
SSSSSS
SSSSSS
SS
SS
SS
SS
SSSSSSSS
SSSSSSSS

```

MA
Sy

\$S
\$C
AR
AU
BL
CH
CH
CH
CH
CH
CH
CH
CH
CH
CH
CH
CN
CR
DA
DA
DA
DA
DB
DC
DC
DC
DD
DE
DE
DG
DI
DI
DL
DM
DM
DO
DO
DO
DP
DP
DP
DS
DS
DS
DT
DU
DU
DU
DU
DU
DU

| | | |
|-----|-----|---|
| (2) | 76 | DECLARATIONS |
| (3) | 116 | MCALL EXPLICIT MACRO CALL ROUTINE |
| (4) | 151 | MAC\$IMPLMCALL IMPLICIT MACRO CALL ROUTINE |
| (5) | 266 | MAC\$GET_MLB_LIN READ LINE FROM MACRO LIBRARY |
| (6) | 293 | LIBRY PROCESS .LIBRARY DIRECTIVE |
| (7) | 377 | MAC\$SYSLIB_SET SET UP THE SYSTEM LIBRARY |
| (8) | 425 | ALLOCATE MCF BLOCK |


```
0000 1      .TITLE  MAC$MACLIB      MACRO LIBRARY PROCESSOR
0000 2      .IDENT  'V04-000'
0000 3
0000 4
0000 5      *****
0000 6      *
0000 7      *  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 8      *  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 9      *  ALL RIGHTS RESERVED.
0000 10     *
0000 11     *  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 12     *  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 13     *  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 14     *  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 15     *  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 16     *  TRANSFERRED.
0000 17     *
0000 18     *  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 19     *  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 20     *  CORPORATION.
0000 21     *
0000 22     *  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 23     *  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 24     *
0000 25     *
0000 26     *****
0000 27
0000 28
0000 29     ++
0000 30     FACILITY:      VAX MACRO ASSEMBLER OBJECT LIBRARY
0000 31
0000 32     ABSTRACT:
0000 33
0000 34     The VAX-11 MACRO assembler translates MACRO-32 source code into object
0000 35     modules for input to the VAX-11 LINKER.
0000 36
0000 37     ENVIRONMENT:  USER MODE
0000 38
0000 39     AUTHOR:  Benn Schreiber, CREATION DATE: 24-AUG-78
0000 40
0000 41     MODIFIED BY:
0000 42
0000 43     V03.01  MTR0016      Mike Rhodes      07-Jun-1982
0000 44     Add logic to MAC$IMPLMCALL to save/modify/restore the
0000 45     listing directive for listing macro definitions. This
0000 46     corrects a problem where the .NOSHOW MD directive caused
0000 47     the listing to lose the invocation lines or other assorted
0000 48     parts when there were no macros explicitly defined in the
0000 49     program text and the macros were being defined from a library.
0000 50     Also included are fixes for broken branch destinations.
0000 51
0000 52     V02.17  MTR0001      Mike Rhodes      09-Nov-1981
0000 53     Add default name descriptor use in opening the system
0000 54     library STARLET.
0000 55
0000 56     V02.16  PCG0007      P. George        20-Aug-1981
0000 57     Report .LIBRARY errors in listing and with exit status.
```

| | | |
|------|----|---|
| 0000 | 58 | : |
| 0000 | 59 | : |
| 0000 | 60 | : |
| 0000 | 61 | : |
| 0000 | 62 | : |
| 0000 | 63 | : |
| 0000 | 64 | : |
| 0000 | 65 | : |
| 0000 | 66 | : |
| 0000 | 67 | : |
| 0000 | 68 | : |
| 0000 | 69 | : |
| 0000 | 70 | : |
| 0000 | 71 | : |
| 0000 | 72 | : |
| 0000 | 73 | : |
| 0000 | 74 | : |

| | | | |
|--------|--|--------------|-------------|
| V01.15 | RN0023 | R. Newland | 3-Nov-1979 |
| | New message codes to get error messages from system message file. | | |
| V01.14 | RN0011 | R. Newland | 11-Sep-1979 |
| | New librarian support | | |
| V01.11 | 004 | B. Schreiber | 10-JAN-1979 |
| | Correct "illegal block size" error if macro library has MNT entries allocated but none used. | | |
| V01.12 | 008 | B. Schreiber | 23-JAN-1979 |
| | Clear index number of pages for SYSLIB FDB on setup. | | |
| V01.13 | RN0003 | R. Newland | 18-Feb-1979 |
| | Correct initial values of macro library binary chop routine | | |

[illegible]

```
0000 76 .SBTTL DECLARATIONS
0000 77 :
0000 78 : INCLUDE FILES:
0000 79 :
0000 80 :
0000 81 :
0000 82 : MACROS:
0000 83 :
0000 84 :
0000 85 $MAC_CTLFLGDEF ;DEFINE CONTROL FLAGS
0000 86 $MAC_GRAMMARDEF ;DEFINE TERMINAL GRAMMAR SYMBOLS
0000 87 $MAC_MLFDEF ; Define MLF offsets
0177 88 $MAC_GENVALDEF ;DEFINE GENERAL VALUES
0177 89 $MAC_INTCODEF ;DEFINE INT. FILE CODES
0177 90 $MAC_SYMBLKDEF ;DEFINE SYMBOL BLOCK OFFSETS
0000 91 $NAMDEF ;DEFINE NAM OFFSETS
0000 92 $RABDEF ;DEFINE RAB OFFSETS
0000 93 $RMEDEF ;DEFINE RME
0000 94 $FABDEF ;DEFINE FAB OFFSETS
0000 95 $MACMSGDEF ; Define message codes
0000 96 :
0000 97 :
0000 98 : LOCAL DATA
0000 99 :
0000 100 :
00000000 101 .PSECT MAC$RO_DATA,NOEXE,NOWRT,GBL, LONG
0000 102 :
0000 103 MLF_ARGLIST:
0000 104 $ARGLST 2,MLF_SIZE,MAC$GL_BASEADDR
000C 105 :
000C 106 MLF_SIZE:
00000177 000C 107 .LONG MLF$K_BLKSI2
0010 108 :
00000000 109 .PSECT MAC$RW_DATA,NOEXE, LONG
0000 110 :
0000 111 MAC$TMP$SYMS: ; Temp sym descriptor (used for
00000001'00000000 0000 112 .LONG 0,MAC$AB_T$PSYM+1 ; LBR$LOOKUP_KEY calls)
0008 113 :
00000000 114 .PSECT MAC$RO_CODE_MAC,NOWRT,GBL, LONG
```



```
0000 116 .SBTTL MCALL EXPLICIT MACRO CALL ROUTINE
0000 117
0000 118 :++
0000 119 : FUNCTIONAL DESCRIPTION:
0000 120 :
0000 121 : THIS ROUTINE IS CALLED WHEN THE .MCALL DIRECTIVE IS FOUND.
0000 122 : ANY MACRO NAMES FOLLOWING THE DIRECTIVE THAT ARE NOT DEFINED
0000 123 : ARE DEFINED BY DOING IMPLICIT MACRO CALLS ON THEM.
0000 124 :
0000 125 :--
0000 126
0000 127 MCALL::
50 0000'CF 01 C3 0000 128 SUBL3 #1,W^MAC$GL_LINEPT,RO :DIRECTIVE = KMCALL
0000'CF 50 D0 0006 129 MOVL RO,W^MAC$GL_ERRPT :SET UP ERROR POINTER
FFF2' 30 000B 130 BSBW MAC$SYMSCNUP :SCAN A MACRO NAME
08 50 E8 000E 131 BLBS RO,10$ :BRANCH IF WE FOUND ONE
0011 132 $MAC_ERR DIRSYNX : Else directive syntax error
FFE7' 31 0016 133 BRW MAC$ERRORPT :ISSUE ERROR AND RETURN
53 0000'CF 9E 0019 134 10$: MOVAB W^MAC$AL_UMCHSHTB,R3 :LOOK UP NAME IN MACRO HASH TABLE
FFDF' 30 001E 135 BSBW MAC$SRCSYMTAB :IF IT IS THERE WE CAN SAVE SOME TIME
OF 50 E8 0021 136 BLBS RO,20$ :BRANCH IF FOUND
21 10 0024 137 BSBB MAC$IMPLMCALL :NO--DO AN IMPLICIT MACRO CALL
0D 58 D1 0026 138 CMPL R8,#MACTXT :DID WE FIND IT?
08 13 0029 139 BEQL 20$ :IF EQL YES
002B 140 $MAC_ERR CANTLOCMAC : No--set message code
FFCD' 30 0030 141 BSBW MAC$ERRORPT :ISSUE MESSAGE TO PASS 2
FFCA' 30 0033 142 20$: BSBW MAC$SKIPSP :SKIP SPACES
2C 5A 91 0036 143 CMPB R10,#^A/,/ :STOP ON A COMMA?
06 12 0039 144 BNEQ 30$ :IF NEQ NO
FFC2' 30 003B 145 BSBW MAC$GETCHR :YES--SKIP IT
FFBF' 30 003E 146 BSBW MAC$SKIPSP :THEN SKIP SPACES
0D 5A 91 0041 147 30$: CMPB R10,#CR :GET TO EOL?
BA 12 0044 148 BNEQ MCALL :IF NEQ NO--KEEP GOING
05 0046 149 RSB :YES--MCALL IS DONE
```

```
0047 151 .SBTTL MAC$IMPLMCALL IMPLICIT MACRO CALL ROUTINE
0047 152
0047 153 :++
0047 154 : FUNCTIONAL DESCRIPTION:
0047 155 :
0047 156 : THIS ROUTINE IS CALLED EITHER BY 'MCALL' OR BY THE MAC$SYMBOL
0047 157 : ROUTINE WHEN AN UNIDENTIFIED NAME IS DETECTED. THE IN-CORE
0047 158 : INDICES OF ALL 'KNOWN' MACRO LIBRARIES ARE SEARCHED STARTING
0047 159 : WITH THE LAST ONE SPECIFIED. THE FOLLOWING METHOD IS USED:
0047 160 :
0047 161 : 1) THE IN-CORE INDEX OF EACH MLB IS SEARCHED USING
0047 162 : A MATCHC INSTRUCTION. IF THE NAME IS FOUND, THE
0047 163 : MLB IS OPENED AND THE MACRO IS DEFINED.
0047 164 :
0047 165 : 2) IF THE NAME IS NOT FOUND IN ANY OF THE MACRO
0047 166 : LIBRARIES, A TOKEN OF 'ERR03' IS RETURNED.
0047 167 :
0047 168 :--
0047 169
0047 170 MAC$IMPLMCALL::
0047 171 PUSH R7 ;SAVE R7
56 0000'CF 9E 0049 172 MOVAB W^MAC$AB_TMP$SYM,R6 ;POINT TO THE NAME WE ARE LOOKING FOR
0000'CF 66 9B 004E 173 MOVZBW (R6),W^MAC$TMP$SYMDS ; Set up descriptor for macro name
57 0000'CF 9E 0053 174 MOVAB W^MAC$GL_MLB_QUE,R7 ;POINT TO MLB QUE HEADER
0058 175
0058 176 NEXT_MLB:
0058 177 MOVL (R7),R7 ;LINK TO NEXT MLB FDB
0058 178 CMPL R7,#MAC$GL_MLB_QUE ;ARE WE DONE?
0062 179 BNEQ 10$ ;IF NEQ NO
0064 180 POPL R7 ;YES--RESTORE R7
58 03 8ED0 0067 181 MOVL #ERR03,R8 ;RETURN ERROR TOKEN
006A 182 RSB
006B 183
006B 184 : Call librarian procedure to look up macro name
006B 185
006B 186 10$:
006B 187 PUSHAB W^MAC$GL_TXT$RFA ; Address to store text RFA
006F 188 PUSHAB W^MAC$TMP$SYMDS ; Address of name descriptor
0073 189 PUSHAB MLF$C_INDEX(R7) ; Address of control table index
0076 190 CALLS #3,G^C$R$LOOKUP_KEY ; Look-up macro name
007D 191 BLBC R0,NEXT_MLB ; Not found if LBC
0080 192 MOVL R7,W^MAC$GL_MLF$PTR ; Save MLF pointer
0085 193
0085 194 BSBW MAC$ALL_1_PAGE ;ALLOCATE A VIRTUAL PAGE
0088 195 PUSH R0 ;SAVE ITS ADDRESS
60 0000'CF 0086 8F 28 008A 196 MOV C3 #LST$K_BUF$SZ,W^MAC$AB_LIN$BF,(R0) ;COPY CURRENT LINE OUT
83 0000'CF DO 0092 197 MOVL W^MAC$GL_LINENUM,(R3)+ ;SAVE CURRENT STATE
83 6B FFFFFFFF 8F CB 0097 198 BICL3 #C<FLG$M_CONT>,(R11),(R3)+ ;SAVE CONTINUATION STATE
83 83 5A DO 009F 199 MOVL R10,(R3)+ ;SAVE CURRENT CHARACTER
83 0000'CF DO 00A2 200 MOVL W^MAC$GL_LIN$PT,(R3)+ ;SAVE LINE POINTER
83 0000'CF DO 00A7 201 MOVL W^MAC$GL_LIN$LN,(R3)+ ;SAVE LINE LENGTH
FF51' 30 00AC 202 BSBW MAC$ALL_1_PAGE ;ALLOCATE AN INPUT BLOCK
60 0000'CF DO 00AF 203 MOVL W^MAC$GL_INPUTP,(R0) ;LINK TO LAST INPUT BLOCK
0000'CF 80 DE 00B4 204 MOVAL (R0)+,W^MAC$GL_INPUTP ;MAKE NEW BLOCK CURRENT BLOCK
80 80 D4 00B9 205 CLRL (R0)+ ;CLEAR TEXT LINK
80 017F'CF 9E 00BB 206 MOVAB W^MAC$GET_MLB_LIN,(R0)+ ;SET NEW LINE ROUTINE
00C0 207 :
```



```
00C0 208 : THE REST OF THE INPUT BLOCK IS NOT USED. HENCE IT IS NOT INITIALIZED.
00C0 209 :
5A 0D 9A 00C0 210 MOVZBL #CR,R10 ;FORCE READING OF NEW LINE
FF3A' 30 00C3 211 BSBW MAC$GETCHR ;GET FIRST CHARACTER OF MACRO
FF37' 30 00C6 212 BSBW MAC$SYMSCNUP ;SCAN FIRST SYMBOL
03 50 E8 00C9 213 BLBS RO,20$ ; Branch if symbol was found
00A5 31 00CC 214 BRW MAC_LIB_FMT_ERR ; ELSE If not found then error
53 00000000'EF 9E 00CF 215 20$: MOVAB MAC$AL_PFMHSHTB,R3 ;LOOK IN PERMANENT SYMBOL TABLE
FF27' 30 00D6 216 BSBW MAC$SRC$SYMTAB ;...
03 50 E8 00D9 217 BLBS RO,30$ ; Branch if symbol was found
0095 31 00DC 218 BRW MAC_LIB_FMT_ERR ; ELSE, If not found then error
50 8F 0B A1 91 00DF 219 30$: CMPB SYMB_TOKEN(R1),#KMACRO ;WAS IT ".MACRO"?
03 13 00E4 220 BEQL 40$ ; If EQL yes.
008B 31 00E6 221 BRW MAC_LIB_FMT_ERR ; If NEQ no -- bad library
00000005'GF DD 00E9 222 40$: PUSHL G^LST$G_MACRODEF+SYMSL_VAL ;SAVE THE LISTING STATE,
0005'CF FF 8F 98 00EF 223 CVTBL #-1,W^LST$G_MACRODEF+SYMSL_VAL ;ENABLE LISTING MACRO DEF's.
00F5 224 $INTOUT_LW INT$ SETLONG,- ;INDICATE THE (POSSIBLE) SWITCH...
00F5 225 <G^LST$G_MACRODEF+SYMSL_VAL,#LST$G_MACRODEF+SYMSL_VAL> ;...TO PASS
FEF6' 30 0107 226 BSBW MACRO ; Define the MACRO
010A 227 :
010A 228 : NOW RESTORE THINGS TO PRE-MLB READING STATE
010A 229 :
00000005'GF 8ED0 010A 230 POPL G^LST$G_MACRODEF+SYMSL_VAL ;RETRIEVE THE PRIOR LISTING STATE
0111 231 $INTOUT_LW INT$ SETLONG,- ;INDICATE THE (POSSIBLE) SWITCH...
0111 232 <G^LST$G_MACRODEF+SYMSL_VAL,#LST$G_MACRODEF+SYMSL_VAL> ;...TO PASS
0000'CF D6 0123 233 INCL W^MAC$GL_MLB MDF ;COUNT MACRO DEFINED FROM MACRO LIBRARY
08 A7 D6 0127 234 INCL MLF$G_MDEF(R7) ;ALSO COUNT IN MLB FDB
50 0000'CF D0 012A 235 MOVL W^MAC$GL_INPUTP,RO ;POINT TO THE INPUT BLOCK
0000'CF 60 D0 012F 236 MOVL (RO),W^MAC$GL_INPUTP ;RESTORE LAST INPUT BLOCK
FEC9' 30 0134 237 BSBW MAC$DEA_1_PAGE ;DEALLOCATE INPUT BLOCK
50 6E D0 0137 238 MOVL (SP),RO ;GET ADDRESS OF SAVED LINE BLOCK
0000'CF 60 0086 8F 28 013A 239 MOVC3 #LST$K_BUFSIZ,(RO),W^MAC$AB_LINEBF ;RESTORE INPUT LINE
0000'CF 81 D0 0142 240 MOVL (R1)+,W^MAC$GL_LINENUM ;RESTORE LINE NUMBER
81 D5 0147 241 TSTL (R1)+ ;CHECK CONT FLAG
06 13 0149 242 BEQL 50$ ;IF EQL CLEAR IT
06 6B 03 E3 014B 243 BBBS #FLG$V_CONT,(R11),60$ ;NO--SET IT
04 11 014F 244 BRB 60$
00 6B 03 E5 0151 245 50$: BBCC #FLG$V_CONT,(R11),60$ ;CLEAR CONT FLAG
5A 81 D0 0155 246 60$: MOVL (R1)+,R10 ;RESTORE CURRENT CHARACTER
0000'CF 81 D0 0158 247 MOVL (R1)+,W^MAC$GL_LINEPT ;RESTORE LINE POINTER
0000'CF 81 D0 015D 248 MOVL (R1)+,W^MAC$GL_LINELN ;RESTORE LINE LENGTH
50 8ED0 0162 249 POPL RO ;RESTORE ADDRESS OF SAVE PAGE
57 8ED0 0165 250 POPL R7 ;RESTORE R7
58 0D D0 0168 251 MOVL #MACTXT,R8 ;RETURN TOKEN FOR MACRO
FE92' 30 016B 252 BSBW MAC$DEA_1_PAGE ;DEALLOCATE PAGE
51 0000'CF D0 016E 253 MOVL W^MAC$GL_MACPTR,R1 ;RETURN PTR TO MNB IN R1
05 0173 254 RSB ;RETURN
0174 255 :
0174 256 :
0174 257 : MACRO LIBRARY FORMAT ERROR. THE CODE TO REPORT THE ERROR IN PASS 2
0174 258 : IS EMITTED TO THE INTERMEDIATE FILE, AND THE ASSEMBLY IS TERMINATED.
0174 259 : PASS 2 IS THEN EXECUTED TO REPORT THE ERRORS.
0174 260 :
0174 261 MAC_LIB_FMT_ERR:
0174 262 $MAC_ERR MACLIBFMTER ; Get the message code
FE84' 30 0179 263 BSBW MAC$ERRORPT ;REPORT ERROR TO PASS 2
FE81' 31 017C 264 BRW MAC$ABORT_PASS1 ;ABORT PASS 1
```

```
017F 266 .SBTTL MAC$GET_MLB_LIN READ LINE FROM MACRO LIBRARY
017F 267
017F 268 :++
017F 269 : FUNCTIONAL DESCRIPTION:
017F 270 :
017F 271 : THIS ROUTINE IS CALLED BY MAC$GETCHR WHEN IT IS TIME TO
017F 272 : READ ANOTHER MACRO DEFINITION LINE FROM A MACRO LIBRARY.
017F 273 : THE LINE IS PLACED IN THE INPUT BUFFER MAC$AB_LINELN.
017F 274 :
017F 275 :--
017F 276
017F 277 MAC$GET_MLB_LIN::
51 0000'CF D0 017F 278 -MOVE W^MAC$GL_MLFPTR,R1 ; Get current MLF pointer
0000'CF 9F 0184 279 PUSHAB W^MAC$GL_LINELN ; Address to store line length
0000'CF 9F 0188 280 PUSHAB W^MAC$GL_LINEBFD ; Address of buffer descriptor
14 A1 9F 018C 281 PUSHAB MLF$L CTINDEX(R1) ; Address of control table index
00000000'GF 03 FB 018F 282 CALLS #3,G^C$R$GET RECORD ; Get record
DB 50 E9 0196 283 BLBC R0,MAC LIB FMT ERR ; IF LBC THEN LIBRARY ERROR
0000'CF D6 0199 284 10$: INCL W^MAC$GL_MLB_GET ; COUNT ANOTHER MLB GET
50 0000'CF D0 019D 285 MOVL W^MAC$GL_LINELN,R0 ; Get line length
DB 13 01A2 286 BEQL MAC$GET_MLB_LIN ; SKIP NULL LINES
51 0000'CF 9E 01A4 287 MOVAB W^MAC$AB_LINEBF,R1 ; POINT TO THE LINE BUFFER
0000'CF 51 D0 01A9 288 MOVL R1,W^MAC$GL_LINEPT ; SET UP THE LINE POINTER
0000'CF 51 D0 01AE 289 MOVL R1,W^MAC$GL_ERRPTX ; AND THE ERROR TOKEN POINTER
6041 0D 90 01B3 290 MOVB #CR,(R0)[R1] ; STORE CR AT END OF LINE
05 01B7 291 RSB
```



```
01B8 293 .SBTTL LIBRY PROCESS .LIBRARY DIRECTIVE
01B8 294
01B8 295 :++
01B8 296 : FUNCTIONAL DESCRIPTION:
01B8 297 :
01B8 298 : THIS ROUTINE IS CALLED TO PROCESS THE .LIBRARY DIRECTIVE.
01B8 299 : THE FILENAME WITHIN THE DELIMITERS IS SCANNED. THE FILE
01B8 300 : IS THEN OPENED, AND AN IN-CORE INDEX IS CREATED. THE FDB
01B8 301 : FOR THE NEW MLB IS THEN ADDED TO THE FRONT OF THE MLB QUEUE.
01B8 302 :
01B8 303 :--
01B8 304
01B8 305 LIBRY::
01B8 306 PUSHL R8 ;DIRECTIVE = KLIBRARY
01B8 307 BSBW MAC$SKIPSP ;SAVE R8
01B8 308 CMPB R10,#CR ;FIND THE DELIMITER
01B8 309 BNEQ 10$ ;DID WE GET TO END OF LINE?
01B8 310 5$: $MAC_ERR DIRSYN ;IF NEQ NO
01B8 311 BRB 40$ ; Yes-get message code
01B8 312 10$: MOVB R10,R6 ;ISSUE ERROR AND RETURN
01B8 313 MOVAB W*MAC$AB_TMPBUF+8,R8 ;COPY THE DELIMITER
01B8 314 ; Point to temp buffer to accumalate name
01B8 315 BISL2 #FLG$M_ALLCHR,(R11) ; (a descriptor will be formed at start)
01B8 316 ;PASS SEMI-COLONS IN CASE SOME
01B8 317 ;TURKEY PUTS A VERSION NUMBER ON
01B8 318 20$: BSBW MAC$GETCHR ;HIS LIBRARY FILE NAME
01B8 319 CMPB R10,R6 ;GET A CHARACTER OF FILENAME
01B8 320 BEQL 50$ ;END OF FILENAME?
01B8 321 CMPB R10,#CR ;IF EQL YES
01B8 322 BEQL 30$ ;BAD ENDING?
01B8 323 MOVB R10,(R8)+ ;IF EQL YES
01B8 324 BRB 20$ ;NO--STORE CHAR OF FILENAME
01B8 325 ;CONTINUE LOOPING FOR NAME
01B8 326 : HERE IF CR BEFORE DELIMITER
01B8 327 :
01B8 328 30$: $MAC_ERR UNTERMARG ; Set message code
01B8 329 40$: POPL R8 ;RESTORE R8
01B8 330 BICL2 #FLG$M_ALLCHR,(R11) ;CLEAR ALLCHR FLAG
01B8 331 BRW MAC$ERRORPT ;ISSUE ERROR AND RETURN
01B8 332 :
01B8 333 : HERE WHEN DELIMITER SEEN
01B8 334 :
01B8 335 50$: BICL2 #FLG$M_ALLCHR,(R11) ;CLEAR ALLCHR FLAG
01B8 336 BSBW MAC$GETCHR ;SKIP OVER DELIMITER
01B8 337 BSBW MAC$SKIPSP ;SKIP SPACES
01B8 338 CMPB R10,#CR ;STOP ON EOL?
01B8 339 BNEQ 5$ ;IF NEQ NO--LOSE
01B8 340 SUBL3 #MAC$AB_TMPBUF+8,R8, - ; form descriptor to library
01B8 341 W*MAC$AB_TMPBUF ; file name
01B8 342 MOVAB W*MAC$AB_TMPBUF+8, -
01B8 343 W*MAC$AB_TMPBUF+4
01B8 344 BSBW MAC$ALL_MLB_MLF ; Get memory block for MLF
01B8 345 :
01B8 346 : Call librarian procedure to initialise library
01B8 347 :
01B8 348 PUSHAB MLF$X_NAMBLK(R6) ; Address of NAM block
01B8 349 PUSHAB W*MAC$GL_LIBTYPE ; Address of type = MLB
```



```
0000'CF 9F 021D 350      PUSHAB W^MAC$GL_LIBFUNC      : Address of function = READ
      14 A6 9F 0221 351      PUSHAB MLF$SL_CTINDEX(R6) : Address of control table index
00000000'GF 04 FB 0224 352      CALLS #4,G^C$INI_CONTROL : Initialise library
      22 50 E9 022B 353      BLBC R0,55$ : Error if LBC
      022E 354 :
      022E 355 : Call Librarian procedure to open library file
      022E 356 :
      0C A6 9F 022E 357      PUSHAB MLF$Q_FNAMDS(R6) : Address of resultant length
      0C A6 9F 0231 358      PUSHAB MLF$Q_FNAMDS(R6) : Address of resultant descriptor
      00 DD 0234 359      PUSHL #0 : No related file name
      0000'CF 9F 0236 360      PUSHAB W^MAC$MLB_DEFNAM : Address of default name
      00 DD 023A 361      PUSHL #0 : No create options
      0000'CF 9F 023C 362      PUSHAB W^MAC$AB_TMPBUF : Address of file name
      14 A6 9F 0240 363      PUSHAB MLF$SL_CTINDEX(R6) : Address of control table index
00000000'GF 07 FB 0243 364      CALLS #7,G^C$OPEN : Open library file
      58 8ED0 024A 365      POPL R8 : RESTORE R8
      14 50 E8 024D 366      BLBS R0,60$ : BRANCH IF GOOD OPEN
      0250 367 55$:
      56 DD 0250 368      PUSHL R6 : Address of MLF
00000000'EF 01 FB 0252 369      CALLS #1,MAC$ERR_LBROPEN : Report library open error
      0259 370      $MAC_ERR_MLBOPNERR : REPORT ERROR WITHIN LISTING
      FD9F' 30 025E 371      BSBW MAC$ERRORPT : AND EXIT WITH ERROR STATUS
      FD9C' 31 0261 372      BRW MAC$ABORT_PASS1 : GO ABORT PASS 1
      0000'CF 66 OE 0264 373 60$:      INSQUE (R6),W^MAC$GL_MLB_QUE : INSERT IN MLB QUEUE
      0000'CF D6 0269 374      INCL W^MAC$GL_MLB_CNT : Increment macro library count
      05 026D 375      RSB
```

```
026E 377 .SBTTL MAC$SYSLIB_SET SET UP THE SYSTEM LIBRARY
026E 378
026E 379 :++
026E 380 : FUNCTIONAL DESCRIPTION:
026E 381 :
026E 382 : THIS ROUTINE IS CALLED AT MACRO-32 INITIALIZATION TO ENSURE
026E 383 : THAT THE SYSTEM LIBRARY EXISTS, AND CREATES AN IN-CORE INDEX
026E 384 : OF IT. THE MACRO LIBRARY QUEUE IS ALSO INITIALIZED.
026E 385 :
026E 386 :--
026E 387
026E 388 MAC$SYSLIB SET::
50 0000'CF 9E 026E 389 MOVAB W^MAC$GL_MLB_QUE,R0 ;INIT THE MLB QUEUE
60 60 DE 0273 390 MOVAL (R0),(R0) ;...
60 80 DE 0276 391 MOVAL (R0)+,(R0) ;...
56 0000'CF 9E 0279 392 MOVAB W^MAC$SYSLIB_MLF,R6 ; Point to SYSLIB MLF entry
14 A6 D4 027E 393 CLRL MLF$CTINDEX(R6) ; Clear control table index
OC A6 00FF'6F B0 0281 394 MOVW #MLF$R_RSFNLN,MLF$Q_FNAMDS(R6) ; Initialise file
10 A6 78 A6 DE 0287 395 MOVAL MLF$T_FNAM(R6),MLF$Q_FNAMDS+4(R6) ; name descriptor
028C 396 :
028C 397 : Call Librarian procedure to initialise library
028C 398 :
18 A6 9F 028C 399 PUSHAB MLF$X_NAMBLK(R6) ; Address of NAM block
0000'CF 9F 028F 400 PUSHAB W^MAC$GL_LIBTYPE ; Address of type = MLB
0000'CF 9F 0293 401 PUSHAB W^MAC$GL_LIBFUNC ; Address of function = READ
14 A6 9F 0297 402 PUSHAB MLF$CTINDEX(R6) ; Address of control table index
00000000'GF 04 FB 029A 403 CALLS #4,G^C[BR$INI_CONTROL ; Initialise library
2B 50 E9 02A1 404 BLBC R0,10$ ; Error if LBC
02A4 405 :
02A4 406 : Call Librarian procedure to open library file
02A4 407 :
OC A6 9F 02A4 408 PUSHAB MLF$Q_FNAMDS(R6) ; Address of resultant length
OC A6 9F 02A7 409 PUSHAB MLF$Q_FNAMDS(R6) ; Address of resultant descriptor
DD DD 02AA 410 PUSHBL #0 ; No related file name
00000000'GF 9F 02AC 411 PUSHAB G^MAC$SYSLIB_DFN ; Address of default name descriptor
DD DD 02B2 412 PUSHBL #0 ; No create options
0000'CF 9F 02B4 413 PUSHAB W^MAC$SYSLIB_FNM ; Address of file name descriptor
14 A6 9F 02B8 414 PUSHAB MLF$CTINDEX(R6) ; Address of control table index
00000000'GF 07 FB 02BB 415 CALLS #7,G^C[BR$OPEN ; Open library file
OA 50 E9 02C2 416 BLBC R0,10$ ; IF LBC THEN ERROR
0000'CF 66 OE 02C5 417 INSQUE (R6),W^MAC$GL_MLB_QUE ; INSERT IN MLB QUEUE
0000'CF D6 02CA 418 INCL W^MAC$GL_MLB_CNT ; Increment macro library count
05 05 02CE 419 RSB
02CF 420 10$:
56 DD 02CF 421 PUSHBL R6 ; Address of MLF
00000000'EF 01 FB 02D1 422 CALLS #1,MAC$ERR_LBROPEN ; Report library open error
FD25' 31 02DB 423 BRW MAC$LAST_CHANCE ; and go die
```

```
02DB 425 .SBTTL ALLOCATE MLF BLOCK
02DB 426
02DB 427 : Functional description:
02DB 428
02DB 429 : This routine is called to allocate a block of memory for
02DB 430 : a MLF (Macro Library File) entry. The allocated memory block
02DB 431 : is zeroed and the NAM block and file name descriptors initialised.
02DB 432
02DB 433 : Inputs:
02DB 434
02DB 435 : None
02DB 436
02DB 437 : Outputs:
02DB 438
02DB 439 : R6 = Address of MLF
02DB 440
02DB 441 :--
02DB 442
02DB 443 MAC$ALL_MLB_MLF::
02DB 444 CALCB MLF_ARGLIST,G^LIB$GET_VM ; Get memory block
02DB 445 BLBC R0,T0$ ; Error if LBC
02DB 446 MOVL W^MAC$GL_BASEADDR,R6 ; Get base address of memory block
02DB 447 MOVCS #0,(SP),#0,#MLF$K_BLKSIZE,(R6) ; Clear MLF
02DB 448 MOVW #MLF$K_RSFNLN,MLF$Q_FNAMDS(R6) ; Initialise file name
02DB 449 MOVAL MLF$T_FNAM(R6),MLF$Q_FNAMDS+4(R6) ; descriptor
02DB 450 MOVW #<<NAM$C_BLN>@8+NAM$C_BID>,- ; Identify NAM block
02DB 451 MLF$X_NAMBLK(R6)
02DB 452 RSB
02DB 453 10$:
02DB 454 CALLS #0,W^MAC$ERR_NOMEM_0 ; Report the error
02DB 455 BRW MAC$LAST_CHANCE ; and go die
02DB 456
02DB 457 .END
```

00000000'GF 00000000'EF FA 02DB 444
1F 50 E9 02E6 445
56 0000'CF D0 02E9 446
66 0177 8F 00 6E 00 2C 02EE 447
0C A6 00FF 8F B0 02F6 448
10 A6 78 A6 DE 02FC 449
6002 8F B0 0301 450
18 A6 0305 451
05 0307 452
0308 453
0000'CF 00 FB 0308 454
FCF0' 31 030D 455
0310 456
0310 457

MAC\$MACLIB
Symbol table

MACRO LIBRARY PROCESSOR

D 8

16-SEP-1984 02:07:45 VAX/VMS Macro V04-00
5-SEP-1984 01:48:58 [MACRO.SRC]MACLIB.MAR;1

Page 12
(8)

\$ST1 = 00000002
\$COUNT = 0000003B
ARG\$K_SIZE = 0000003E8
AUD\$K_SIZE = 00000010
BLNK = 00000020
CHRS\$M_COMMA_CR = 00000020
CHRS\$M_ILL_CHR = 00000040
CHRS\$M_NUM_BER = 00000010
CHRS\$M_SPA_MSK = 00000001
CHRS\$M_SYM_CH1 = 00000008
CHRS\$M_SYM_CHR = 00000004
CHRS\$M_SYM_DLM = 00000002
CHRS\$V_COMMA_CR = 00000005
CHRS\$V_CVT_LWC = 00000061
CHRS\$V_ILL_CHR = 00000006
CHRS\$V_NOCVT = 0000007F
CHRS\$V_NUM_BER = 00000004
CHRS\$V_SPA_MSK = 00000000
CHRS\$V_SYM_CH1 = 00000003
CHRS\$V_SYM_CHR = 00000002
CHRS\$V_SYM_DLM = 00000001
CNT = 00000002
CR = 0000000D
DAND = 0000001D
DANGCLS = 00000016
DANGOPN = 00000015
DAT = 00000020
DBUP = 0000002B
DCLS = 00000018
DCOLON = 00000010
DCOMMA = 0000000F
DDIV = 0000001C
DEOL = 0000000B
DEQ = 00000011
DGUP = 0000002C
DINTEGER = 00000022
DIUP = 0000002D
DLUP = 0000002E
DMASK = 00000032
DMINUS = 0000001A
DOPCODE = 0000000E
DOPN = 00000017
DOR = 0000001E
DPC = 00000012
DPLUS = 00000019
DPOUND = 00000021
DSQCLS = 00000014
DSQOPN = 00000013
DSUP = 0000002F
DTIMES = 0000001B
DUPA = 00000023
DUPB = 00000024
DUPC = 00000025
DUPD = 00000026
DUPF = 00000028
DUPM = 00000029
DUPO = 00000027

DUPX = 0000002A
DWUP = 00000030
DXOR = 0000001F
ERR = 00000000
ERR01 = 00000001
ERR02 = 00000002
ERR03 = 00000003
ERR04 = 00000004
ERR05 = 00000005
ERR06 = 00000006
ERR07 = 00000007
ERR08 = 00000008
ERR09 = 00000009
FF = 0000000C
FLG\$M_ALLCHR = 00000001
FLG\$M_BOL = 00000002
FLG\$M_CHKLPND = 00100000
FLG\$M_COMPEXPR = 00000004
FLG\$M_CONT = 00000008
FLG\$M_CRF = 40000000
FLG\$M_CRSEEN = 00000001
FLG\$M_DATRPT = 00000010
FLG\$M_DBGOUT = 00004000
FLG\$M_DLMSTR = 00008000
FLG\$M_ENDMCH = 00000020
FLG\$M_EVALEXPR = 00000040
FLG\$M_EXPOPT = 00000080
FLG\$M_EXTERR = 00010000
FLG\$M_EXTWRN = 00020000
FLG\$M_FIRSTLN = 00000200
FLG\$M_IFSTAT = 00800000
FLG\$M_IIF = 00400000
FLG\$M_INSERT = 00000100
FLG\$M_IRPC = 20000000
FLG\$M_LEXOP = 00000002
FLG\$M_LSTXST = 00000200
FLG\$M_MAC2COL = 00000800
FLG\$M_MACL = 00000800
FLG\$M_MACLTB = 08000000
FLG\$M_MACTXT = 00010000
FLG\$M_MEBLST = 00001000
FLG\$M_MOREARG = 00002000
FLG\$M_MOREINP = 00000008
FLG\$M_NEWPND = 00000400
FLG\$M_NOREF = 01000000
FLG\$M_NTTYPEPC = 00000020
FLG\$M_NULCHR = 00040000
FLG\$M_OBJXST = 00200000
FLG\$M_OPNDCHK = 00000100
FLG\$M_OPRND = 00002000
FLG\$M_OPTVFLIDX = 00001000
FLG\$M_ORDLST = 00020000
FLG\$M_P2 = 00004000
FLG\$M_RPTIRP = 10000000
FLG\$M_SEQFIL = 02000000
FLG\$M_SKAN = 00008000
FLG\$M_SPECOP = 00000004

MA
VO

MACSMACLIB
Symbol table

MACRO LIBRARY PROCESSOR

E 8

16-SEP-1984 02:07:45 VAX/VMS Macro V04-00
5-SEP-1984 01:48:58 [MACRO.SRC]MACLIB.MAR;1

Page 13
(8)

FLGSM_SPLALL = 04000000
FLGSM_STOIMF = 00040000
FLGSM_SYM2COL = 00000400
FLGSM_TOCLFG = 00080000
FLGSM_UPAFLG = 00000010
FLGSM_UPDFIL = 00000080
FLGSM_UPMARG = 00000040
FLGSM_XCRF = 80000000
FLGSV_ALLCHR = 00000000
FLGSV_BOL = 00000001
FLGSV_CHKLPND = 00000014
FLGSV_COMPEXP = 00000002
FLGSV_CONT = 00000003
FLGSV_CRF = 0000001E
FLGSV_CRSEEN = 00000020
FLGSV_DATRPT = 00000004
FLGSV_DBGOUT = 0000002E
FLGSV_DLIMSTR = 0000002F
FLGSV_ENDMCH = 00000005
FLGSV_EVAL_EXPR = 00000006
FLGSV_EXPORT = 00000007
FLGSV_EXTERR = 00000030
FLGSV_EXTURN = 00000031
FLGSV_FIRSTLN = 00000029
FLGSV_IFSTAT = 00000017
FLGSV_IIF = 00000016
FLGSV_INSERT = 00000008
FLGSV_IRPC = 0000001D
FLGSV_LEXOP = 00000021
FLGSV_LSTXST = 00000009
FLGSV_MAC2COL = 0000002B
FLGSV_MACL = 0000000B
FLGSV_MACLTB = 0000001B
FLGSV_MACTXT = 00000010
FLGSV_MEBLST = 0000000C
FLGSV_MOREARG = 0000002D
FLGSV_MOREINP = 00000023
FLGSV_NEWPND = 0000000A
FLGSV_NOREF = 0000001B
FLGSV_NTTYPEPC = 00000025
FLGSV_NULCHR = 00000032
FLGSV_OBJXST = 00000015
FLGSV_OPNDCHK = 00000028
FLGSV_OPRND = 0000000D
FLGSV_OPTVFLIDX = 0000002C
FLGSV_ORDLST = 00000011
FLGSV_P2 = 0000000E
FLGSV_RPTIRP = 0000001C
FLGSV_SEQFIL = 00000019
FLGSV_SKAN = 0000000F
FLGSV_SPECOP = 00000022
FLGSV_SPLALL = 0000001A
FLGSV_STOIMF = 00000012
FLGSV_SYM2COL = 0000002A
FLGSV_TOCLFG = 00000013
FLGSV_UPAFLG = 00000024
FLGSV_UPDFIL = 00000027

FLGSV_UPMARG = 00000026
FLGSV_XCRF = 0000001F
GOALST = 0000000A
HASHSZ = 0000007F
HYPHEN = 0000002D
ID = 0000000C
INPSK_BUFSIZ = 000003E8
INTSK_BUFSIZ = 000013F4
INTSK_BUFWRN = 00001390
INTS_ADD = 00000001
INTS_AND = 00000002
INTS_ASH = 00000003
INTS_ASN = 0000000C
INTS_AUGPC = 0000000D
INTS_BDST = 0000000E
INTS_CHKL = 0000000F
INTS_DIV = 00000004
INTS_END = 00000010
INTS_EPT = 00000011
INTS_ERR = 00000012
INTS_ETX = 00000013
INTS_FNEWL = 00000014
INTS_ILG = 00000000
INTS_INFO = 0000003A
INTS_LGLAB = 00000015
INTS_MACL = 00000016
INTS_MUL = 00000005
INTS_NEG = 00000006
INTS_NEWL = 00000017
INTS_NEWP = 00000018
INTS_NOT = 00000007
INTS_OP = 00000019
INTS_OR = 00000008
INTS_PRIL = 0000001A
INTS_PRT = 0000001B
INTS_PSECT = 0000001C
INTS_REDEF = 0000001D
INTS_REF = 0000001E
INTS_REST = 0000001F
INTS_SAME = 00000009
INTS_SAVE = 00000020
INTS_SBTTL = 00000021
INTS_SETFLAG = 00000022
INTS_SETLONG = 00000023
INTS_SPIC = 00000024
INTS_SPID = 00000025
INTS_STIB = 00000026
INTS_STIL = 00000028
INTS_STIW = 00000027
INTS_STKEPT = 00000029
INTS_STKG = 0000002A
INTS_STKL = 0000002B
INTS_STKPC = 0000002C
INTS_STKS = 0000002D
INTS_STOB = 00000034
INTS_STOL = 0000002E
INTS_STOW = 00000035

MA
VO

| | |
|-------------|------------|
| INT\$-STRB | = 0000002F |
| INT\$-STRL | = 00000031 |
| INT\$-STRSB | = 00000032 |
| INT\$-STRSW | = 00000033 |
| INT\$-STRW | = 00000030 |
| INT\$-STSB | = 00000036 |
| INT\$-STSW | = 00000037 |
| INT\$-SUB | = 0000000A |
| INT\$-SUME | = 00000039 |
| INT\$-WRN | = 00000038 |
| INT\$-XOR | = 0000000B |
| KADDRESS | = 00000037 |
| KALIGN | = 0000005A |
| KASCIC | = 00000033 |
| KASCID | = 00000078 |
| KASCI1 | = 00000034 |
| KASCI2 | = 00000035 |
| KBLKA | = 0000003F |
| KBLKB | = 00000040 |
| KBLKD | = 00000041 |
| KBLKF | = 00000042 |
| KBLKG | = 0000007E |
| KBLKH | = 0000007F |
| KBLKL | = 00000043 |
| KBLKO | = 00000080 |
| KBLKQ | = 00000044 |
| KBLKW | = 00000045 |
| KBYTE | = 00000038 |
| KCROSS | = 00000079 |
| KDEBUG | = 00000055 |
| KDFLT | = 0000007B |
| KDOUBLE | = 00000039 |
| KDSABL | = 00000056 |
| KENABL | = 00000057 |
| KEND | = 00000076 |
| KENDC | = 0000004E |
| KENDM | = 00000053 |
| KENDR | = 0000004F |
| KENTRY | = 00000058 |
| KERROR | = 00000071 |
| KEVEN | = 0000005B |
| KEXTRN | = 0000005D |
| KFIELD | = 0000003A |
| KFLOAT | = 0000003B |
| KGFLOAT | = 00000081 |
| KGLOBL | = 0000005E |
| KHFLOAT | = 00000082 |
| KIDENT | = 0000006A |
| KIF | = 00000046 |
| KIFF | = 00000048 |
| KIFT | = 00000049 |
| KIFTF | = 0000004A |
| KIIF | = 00000047 |
| KINCLUDE | = 0000005F |
| KIRP | = 0000004B |
| KIRPC | = 0000004C |
| KLIBRARY | = 00000060 |

| | | | |
|-------------------|------------|----|----|
| KLINK | = 00000085 | | |
| KLIST | = 00000061 | | |
| KLONG | = 0000003C | | |
| KMACRO | = 00000050 | | |
| KMCALL | = 00000051 | | |
| KMDELETE | = 00000054 | | |
| KMEXIT | = 00000052 | | |
| KNARG | = 00000063 | | |
| KNCHR | = 00000064 | | |
| KNCROS | = 0000007A | | |
| KNLIST | = 00000062 | | |
| KNTYPE | = 00000074 | | |
| KOCTA | = 00000083 | | |
| KODD | = 0000005C | | |
| KOPDEF | = 00000075 | | |
| KPACKED | = 00000036 | | |
| KPAGE | = 00000065 | | |
| KPRINT | = 00000072 | | |
| KPSECT | = 00000066 | | |
| KQUAD | = 0000003D | | |
| KREF1 | = 0000006D | | |
| KREF16 | = 00000084 | | |
| KREF2 | = 0000006E | | |
| KREF4 | = 0000006F | | |
| KREF8 | = 00000070 | | |
| KREPT | = 0000004D | | |
| KRESTORE | = 00000067 | | |
| KSAVE | = 00000068 | | |
| KSBTTL | = 0000006B | | |
| KSGNB | = 0000007C | | |
| KSGNW | = 0000007D | | |
| KTITLE | = 00000069 | | |
| KVECTOR | = 00000059 | | |
| KWARN | = 00000073 | | |
| KWEAK | = 0000006C | | |
| KWORD | = 0000003E | | |
| KXFER | = 00000077 | | |
| LBR\$GET_RECORD | ***** | X | 05 |
| LBR\$INI_CONTROL | ***** | X | 05 |
| LBR\$LOOKUP_KEY | ***** | X | 05 |
| LBR\$OPEN | ***** | X | 05 |
| LIB\$GET_VM | ***** | X | 05 |
| LIBRY | 000001B8 | RG | 05 |
| LST\$G_MACRODEF | ***** | X | 05 |
| LST\$K_BUF\$IZ | = 00000086 | | |
| LST\$K_L_P_PAGE | = 0000003C | | |
| LST\$K_TITLE_SIZ | = 00000028 | | |
| MAC\$ABORT_PASS1 | ***** | X | 05 |
| MAC\$AB_LINEBF | ***** | X | 05 |
| MAC\$AB_TMPBUF | ***** | X | 05 |
| MAC\$AB_TMP\$YM | ***** | X | 04 |
| MAC\$ALC_1_PAGE | ***** | X | 05 |
| MAC\$ALL_MCB_MLF | 000002DB | RG | 05 |
| MAC\$AL_P\$M\$HTB | ***** | X | 05 |
| MAC\$AL_UMCH\$HTB | ***** | X | 05 |
| MAC\$DEX_1_PAGE | ***** | X | 05 |
| MAC\$ERRORPT | ***** | X | 05 |

MACSMACLIB
Symbol table

MACRO LIBRARY PROCESSOR

G 8

16-SEP-1984 02:07:45 VAX/VMS Macro V04-00
5-SEP-1984 01:48:58 [MACRO.SRC]MACLIB.MAR;1

Page 15
(8)

| | | | | | | |
|--------------------|------------|----|----|------------------|---|----------|
| MACSERR_LBROPEN | ***** | X | 05 | OBJ\$K_BUF\$IZ | = | 00000200 |
| MACSERR_NOMEM_0 | ***** | X | 05 | OPF\$M_LASTOPR | = | 00002000 |
| MAC\$GETCHR | ***** | X | 05 | OPF\$M_OPTEXP | = | 00001000 |
| MAC\$GET_MLB_LIN | 0000017F | RG | 05 | OPF\$V_LASTOPR | = | 0000000D |
| MAC\$GL_BASEADDR | ***** | X | 03 | OPF\$V_OPTEXP | = | 0000000C |
| MAC\$GL_ERRPT | ***** | X | 05 | PSC\$B_NAME | | 00000004 |
| MAC\$GL_ERRPTX | ***** | X | 05 | PSC\$B_SEG | | 0000000C |
| MAC\$GL_INPUTP | ***** | X | 05 | PSC\$B_UNUSED | | 0000000B |
| MAC\$GL_LIBFUNC | ***** | X | 05 | PSC\$K_BLK\$IZ | | 00000013 |
| MAC\$GL_LIBTYPE | ***** | X | 05 | PSC\$K_NO_OPTNS | = | 0000000A |
| MAC\$GL_LINELN | ***** | X | 05 | PSC\$L_CURLOC | | 0000000F |
| MAC\$GL_LINENUM | ***** | X | 05 | PSC\$L_LINK | | 00000000 |
| MAC\$GL_LINEPT | ***** | X | 05 | PSC\$L_MAXLGTH | | 00000005 |
| MAC\$GL_MACPTR | ***** | X | 05 | PSC\$M_ABS | = | FFFFFFF7 |
| MAC\$GL_MLB_CNT | ***** | X | 05 | PSC\$M_ALIGNFLG | = | 00004000 |
| MAC\$GL_MLB_GET | ***** | X | 05 | PSC\$M_ALLOPTNS | = | 000003FF |
| MAC\$GL_MLB_MDF | ***** | X | 05 | PSC\$M_BYTE | = | 00004000 |
| MAC\$GL_MLB_QUE | ***** | X | 05 | PSC\$M_CON | = | FFFFFFFB |
| MAC\$GL_MLFPTR | ***** | X | 05 | PSC\$M_DEFAULT | = | 000001C8 |
| MAC\$GL_TXTRFA | ***** | X | 05 | PSC\$M_EXE | = | 000000C0 |
| MAC\$GO_LINEBFD\$ | ***** | X | 05 | PSC\$M_GBL | = | 00000010 |
| MAC\$IMPLMCALL | 00000047 | RG | 05 | PSC\$M_LCL | = | FFFFFFEF |
| MAC\$INTOUT_2_LW | ***** | X | 05 | PSC\$M_LIB | = | 00000002 |
| MAC\$LAST_CRANCE | ***** | X | 05 | PSC\$M_LONG | = | 00004800 |
| MAC\$MLB_DEFNAM | ***** | X | 05 | PSC\$M_NOEXE | = | FFFFFFBF |
| MAC\$SKIPSP | ***** | X | 05 | PSC\$M_NOPIC | = | FFFFFFFE |
| MAC\$SRCSYMTAB | ***** | X | 05 | PSC\$M_NORD | = | FFFFFF7F |
| MAC\$SYMSCNUP | ***** | X | 05 | PSC\$M_NOSHR | = | FFFFFFDF |
| MAC\$SYSLIB_DFN | ***** | X | 05 | PSC\$M_NOVEC | = | FFFFFFDF |
| MAC\$SYSLIB_FNM | ***** | X | 05 | PSC\$M_NOWRT | = | FFFFFFEF |
| MAC\$SYSLIB_MLF | ***** | X | 05 | PSC\$M_OVR | = | 00000004 |
| MAC\$SYSLIB_SET | 0000026E | RG | 05 | PSC\$M_PAGE | = | 00006400 |
| MAC\$TMPSYMD\$ | 00000000 | R | 04 | PSC\$M_PIC | = | 00000001 |
| MAC\$CANTLOCMAC | = 007D905A | | | PSC\$M_QUAD | = | 00004C00 |
| MAC\$DIRSYNX | = 007D906A | | | PSC\$M_RD | = | 00000080 |
| MAC\$MACLB\$FMTERR | = 007D913A | | | PSC\$M_REL | = | 00000008 |
| MAC\$MLBOPNERR | = 007D923A | | | PSC\$M_SHR | = | 00000020 |
| MAC\$UNTERMARG | = 007D922A | | | PSC\$M_USR | = | FFFFFFFD |
| MACRO | ***** | X | 05 | PSC\$M_VEC | = | 00000200 |
| MACTXT | = 0000000D | | | PSC\$M_WORD | = | 00004400 |
| MAC_LIB_FMT_ERR | 00000174 | R | 05 | PSC\$M_WRT | = | 00000180 |
| MAC_SUBSYS | = 0000007D | | | PSC\$S_ALIGNMENT | = | 00000004 |
| MACALL | 00000000 | RG | 05 | PSC\$V_ALIGNFLG | = | 0000000E |
| MLF\$K_BLK\$IZ | 00000177 | | | PSC\$V_ALIGNMENT | = | 0000000A |
| MLF\$K_RS\$NLN | = 000000FF | | | PSC\$V_EXE | = | 00000006 |
| MLF\$L_CTINDEX | 00000014 | | | PSC\$V_GBL | = | 00000004 |
| MLF\$L_M\$DEF | 00000008 | | | PSC\$V_LIB | = | 00000001 |
| MLF\$L_QLINK | 00000000 | | | PSC\$V_OVR | = | 00000002 |
| MLF\$Q_FNAMDS | 0000000C | | | PSC\$V_PIC | = | 00000000 |
| MLF\$T_FNAM | 00000078 | | | PSC\$V_RD | = | 00000007 |
| MLF\$X_NAMBLK | 00000018 | | | PSC\$V_REL | = | 00000003 |
| MLF_ARGLIST | 00000000 | R | 03 | PSC\$V_SHR | = | 00000005 |
| MLF_SIZE | 0000000C | R | 03 | PSC\$V_VEC | = | 00000009 |
| NAM\$C_BID | = 00000002 | | | PSC\$V_WRT | = | 00000008 |
| NAM\$C_BLN | = 00000060 | | | PSC\$W_FLAG | | 00000009 |
| NAM\$C_MAXRSS | = 000000FF | | | PSC\$W_OPTIONS | | 0000000D |
| NEXT_MLB | 00000058 | R | 05 | RDX\$V_BINARY | = | 00000000 |

MACSMACLIB
Symbol table

MACRO LIBRARY PROCESSOR

H 8

16-SEP-1984 02:07:45 VAX/VMS Macro V04-00
5-SEP-1984 01:48:58 [MACRO.SRC]MACLIB.MAR;1

Page 16
(8)

| | |
|----------------|------------|
| RDXSV_DECIMAL | = 00000002 |
| RDXSV_DOUBLE | = 00000005 |
| RDXSV_FLOAT | = 00000004 |
| RDXSV_GFLOAT | = 00000006 |
| RDXSV_HEX | = 00000003 |
| RDXSV_HFLOAT | = 00000007 |
| RDXSV_OCTAL | = 00000001 |
| REGS_PC | = 0000000F |
| RRREG | = 00000031 |
| SEMI | = 0000003B |
| STBSK_PG_MISS | = 0000000A |
| SYMSB_NAME | 00000004 |
| SYMSB_SEG | 0000000C |
| SYMSB_TOKEN | 0000000B |
| SYMSK_BLKSI2 | 0000000D |
| SYMSK_MAXLEN | = 0000001F |
| SYMSK_TWOCOL | = 00000010 |
| SYMSL_LINK | 00000000 |
| SYMSL_VAL | 00000005 |
| SYMSM_ABS | = 00000010 |
| SYMSM_ASN | = 00000100 |
| SYMSM_CRFO | = 00002000 |
| SYMSM_DEBUG | = 00000020 |
| SYMSM_DEF | = 00000001 |
| SYMSM_DELMAC | = 00000200 |
| SYMSM_EPT | = 00000200 |
| SYMSM_EXTRN | = 00000008 |
| SYMSM_GLOBL | = 00000004 |
| SYMSM_LOCAL | = 00000040 |
| SYMSM_ODBG | = 00000400 |
| SYMSM_REF | = 00000080 |
| SYMSM_RELPSECT | = 00000800 |
| SYMSM_SUPR | = 00004000 |
| SYMSM_WEAK | = 00000002 |
| SYMSM_XCRF | = 00001000 |
| SYMSV_ABS | = 00000004 |
| SYMSV_ASN | = 00000008 |
| SYMSV_CRFO | = 0000000D |
| SYMSV_DEBUG | = 00000005 |
| SYMSV_DEF | = 00000000 |
| SYMSV_DELMAC | = 00000009 |
| SYMSV_EPT | = 00000009 |
| SYMSV_EXTRN | = 00000003 |
| SYMSV_GLOBL | = 00000002 |
| SYMSV_LOCAL | = 00000006 |
| SYMSV_ODBG | = 0000000A |
| SYMSV_REF | = 00000007 |
| SYMSV_RELPSECT | = 0000000B |
| SYMSV_SUPR | = 0000000E |
| SYMSV_WEAK | = 00000001 |
| SYMSV_XCRF | = 0000000C |
| SYMSW_FLAG | 00000009 |
| TAB | = 00000009 |
| X1 | = 00000400 |
| X2 | = 0000000F |

! Psect synopsis !

| PSECT name | Allocation | PSECT No. | Attributes |
|------------------|------------------|-----------|---|
| . ABS . | 00000000 (0.) | 00 (0.) | NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE |
| . BLANK . | 00000000 (0.) | 01 (1.) | NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE |
| \$ABSS | 00000177 (375.) | 02 (2.) | NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE |
| MAC\$RO_DATA | 00000010 (16.) | 03 (3.) | NOPIC USR CON REL GBL NOSHR NOEXE RD NOWRT NOVEC LONG |
| MAC\$RW_DATA | 00000008 (8.) | 04 (4.) | NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC LONG |
| MAC\$RO_CODE_MAC | 00000310 (784.) | 05 (5.) | NOPIC USR CON REL GBL NOSHR EXE RD NOWRT NOVEC LONG |

! Performance indicators !

| Phase | Page faults | CPU Time | Elapsed Time |
|------------------------|-------------|-------------|--------------|
| Initialization | 30 | 00:00:00.04 | 00:00:01.80 |
| Command processing | 103 | 00:00:00.39 | 00:00:02.31 |
| Pass 1 | 287 | 00:00:06.03 | 00:00:29.58 |
| Symbol table sort | 5 | 00:00:00.98 | 00:00:02.90 |
| Pass 2 | 99 | 00:00:01.29 | 00:00:05.01 |
| Symbol table output | 60 | 00:00:00.26 | 00:00:01.51 |
| Psect synopsis output | 2 | 00:00:00.02 | 00:00:00.03 |
| Cross-reference output | 0 | 00:00:00.00 | 00:00:00.00 |
| Assembler run totals | 588 | 00:00:09.01 | 00:00:43.14 |

The working set limit was 1350 pages.

55114 bytes (108 pages) of virtual memory were used to buffer the intermediate code.

There were 60 pages of symbol table space allocated to hold 1042 non-local and 20 local symbols.

457 source lines were read in Pass 1, producing 24 object records in Pass 2.

17 pages of virtual memory were used to define 16 macros.

! Macro library statistics !

| Macro library name | Macros defined |
|-------------------------------------|----------------|
| \$255\$DUA28:[MACRO.OBJ]MACRO.MLB;1 | 11 |
| \$255\$DUA28:[SYSLIB]STARLET.MLB;2 | 8 |
| TOTALS (all libraries) | 19 |

1142 GETS were required to define 19 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:MACLIB/OBJ=OBJ\$:MACLIB MSRC\$:MACLIB/UPDATE=(ENH\$:MACLIB)+LIB\$:MACRO/LIB

0226 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

